

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

1 1. (Currently Amended) A heuristics analysis tool embodied in a computer-readable
2 storage medium, comprising:
3 a persistent table, having clean data records and key records wherein ~~there is~~ at least one
4 key record is associated with each clean data record, ~~[[said]]~~ each key record having at least one
5 field of data from ~~[[an]]~~ the associated ~~[[said]]~~ clean data record; and
6 ~~associated with said key records~~, heuristic-based routines ~~for automatically generating~~
7 ~~said key records from each to match~~ newly received data ~~record for matching to said clean data~~
8 ~~record~~ records to the key records in the persistent table, the heuristic-based routines to iteratively
9 clean the newly received data records by modifying the newly received data records in response
10 to no match occurring between the received data records and the key records in the persistent
11 table.

1 2. (Currently Amended) The tool as set forth in claim 1 wherein ~~[[said]]~~ at least one of the
2 clean data records includes ~~record is a primary-clean~~ most accurate key record of a plurality of
3 said key records ~~in a set~~ associated with the at least one clean data record.

1 3. (Cancelled)

1 4. (Currently Amended) The tool as set forth in claim 1 wherein each said clean data record
2 is a ~~primary-complete~~ completely clean data file.

1 5. (Original) The tool as set forth in claim 1 further comprising:
2 at least one column recording one or more of said heuristic-based routines that were
3 involved in generating each of said key records.

1 6. (Currently Amended) The tool as set forth in claim 1 further comprising:
2 a time-stamp associated with each said key record in the table wherein said time-stamp is
3 indicative of most recent use.

1 7. (Original) The tool as set forth in claim 1 further comprising:
2 special flags associated with said key records, said flags associated with specific heuristic
3 considerations.

1 8. (Currently Amended) The tool as set forth in claim 7 wherein [[a]] one of the special
2 [[flag]] flags is a quality factor assigned to each said key record.

1 9. (Currently Amended) A data association and cleaning method comprising:
2 storing a plurality of clean data files and, associated with each of said clean data files, at
3 least one indexing record, each said indexing record containing at least one field related to a
4 respective associated clean data file such that said at least one indexing record serves as a pointer
5 to the respective associated said clean data file;
6 comparing ~~incoming an input data records~~ record to the indexing records for obtaining a
7 match, and if the match occurs, assigning said input data record to the respective associated said
8 clean data file ~~associated with a matched indexing record~~;
9 ~~if not obtaining a said~~ the match does not occur, iteratively cleaning the input data record
10 until at least a near-match between said cleaned input data record and [[said]] at least one of the
11 ~~indexing record records~~ is obtained and assigning said cleaned input data record to the one of
12 said clean data files associated with [[a]] the near-matched indexing record; and
13 upon a near match, adding ~~a so-cleaned~~ said cleaned input data record as a new indexing
14 record for [[an]] the associated one of said clean data files, and upon no match, adding said
15 ~~so-cleaned~~ cleaned input data record as a new clean data file with an associated indexing record
16 therefor.

1 10. (Original) The method as set forth in claim 9 wherein said storing is in a displayable
2 format.

1 11. (Original) The method as set forth in claim 10 further comprising:
2 at given intervals, performing a data clean-up on a stored table in said displayable format.

1 12. (Currently Amended) The method as set forth in claim 9 wherein upon said adding said
2 ~~so cleaned~~ cleaned input data record as a new clean data file with an associated indexing record
3 therefor, flagging said new clean data file.

1 13. (Currently Amended) The method as set forth in claim 9, said iteratively cleaning further
2 comprising:

3 ~~upon said not recognizing a match therebetween,~~ cleaning said input data record and
4 storing a first cleaned input data [[set]] record;

5 comparing the first cleaned input data record to ~~each~~ said indexing ~~record~~ records, and

6 upon recognizing a match therebetween, stopping said comparing, and retrieving
7 the associated clean data file for association with said first cleaned input data record,

8 upon not recognizing a match therebetween, re-cleaning said first cleaned input
9 data record, discarding said first cleaned input data record, and storing a subsequently cleaned
10 input data [[set]] record;

11 re-comparing the subsequently cleaned input data set to said indexing ~~record~~ records; and

12 iteratively repeating said re-cleaning and re-comparing until a predetermined phase of
13 cleaning is reached ~~without the~~ and no said match therebetween ~~wherein said a last said~~

14 ~~subsequently cleaned input data is stored~~ is determined, and storing the most recent re-cleaned
15 input data record as a new clean data file.

1 14. (Currently Amended) The method as set forth in claim 13 wherein upon ~~said recognizing~~
2 ~~a match therebetween,~~ reaching the predetermined phase of cleaning with no match, generating a
3 new indexing record ~~is generated~~ for said [[new]] associated clean data file.

1 15. (Currently Amended) A computer memory comprising:
2 computer code means for receiving an input data ~~records~~ record;
3 computer code means for comparing said input data ~~records~~ record to a tabular format set
4 of crude keys;
5 computer code means for returning a clean key associated with one of said crude keys
6 upon a comparing match;
7 computer code means for iterative cleaning of said input data ~~records~~ record upon a
8 no-match return and storing ~~[[an]]~~ the iteratively-generated respective cleaned input data record
9 therefrom;
10 computer code means for re-comparing said iteratively-generated respective cleaned data
11 record to said set of crude keys, ~~searching for said match return~~; and
12 computer code means for creating a new ~~data file~~ crude key from a last said iteratively-
13 generated respective cleaned input data record such that said new ~~data file is also a first~~
14 ~~respective one of said crude keys associated therewith~~ crude key is added to the set of crude
15 keys.

1 16. (Cancelled)

1 17. (Currently Amended) The computer memory as set forth in claim ~~[[16]]~~ 15 wherein said
2 computer code means for generating ~~[[a]]~~ the new crude key has heuristic routines.

1 18. (Currently Amended) The computer memory as set forth in claim 17 further comprising:
2 computer code means for displaying in said tabular format said crude keys and heuristic
3 routines ~~employed in said generating~~.

1 19. (Original) The computer memory as set forth in claim 15 wherein each of said crude
2 keys has an associated pointer to obtain said associated clean key.

1 20. (Currently Amended) The computer memory as set forth in claim [[17]] 19 wherein each
2 of said crude keys points to a cleanest one of a plurality of crude keys associated with a clean
3 data file.

1 21. (Currently Amended) The computer memory as set forth in claim 15 wherein said
2 tabular [[array]] format is a displayable table, further comprising:
3 computer code means including heuristic routines for editing said table.

1 22. (Cancelled)

1 23. (Currently Amended) A method of doing business comprising:
2 storing a database of clean data files for each of a plurality of entities;
3 creating a tabulation of crude keys, each having a pointer to an associated one of said
4 clean data files;
5 receiving ~~periodically~~ a ~~potentially~~ dirty data record related to at least one entity of said
6 plurality of entities;
7 comparing said dirty data record to said [[array]] tabulation;
8 assigning said dirty data record to one of said clean data files if a match is found based on
9 the comparing; [[and]]
10 ~~creating new clean data files from said dirty data when no pointer substantially matching~~
11 ~~said dirty data is found during said comparing~~ cleaning the dirty data record by modifying the
12 dirty data record in response to determining that no match is present based on the comparing; and
13 comparing the cleaned dirty data record to said tabulation.